

Dokumentation zum Workshop

„MeinErstesAddon“

Stand 2014-10-13

RK-Workshop-MeinErstesAddon

Einleitung:

Im Folgenden wird ausführlich beschrieben, wie man selber AddOns programmiert.

Ein **AddOn** ist ein kleines Zusatzprogramm neben der eigentlichen Software von Cockpit XP, welches sich individuell selber programmieren lässt.

Programmierkenntnisse sind sicherlich von Vorteil, aber nicht zwingende eine Voraussetzung.

Mit etwas Mut und gutem Willen kann man relativ schnell die Funktionsweise der Addons Programmierung erlernen und sich dann seine eigenen AddOns selber programmieren

RK-Workshop-MeinErstesAddOn

Inhalt

Vorraussetzungen für den Workshop..... 4
Impressum..... 19

RK-Workshop-MeinErstesAddon

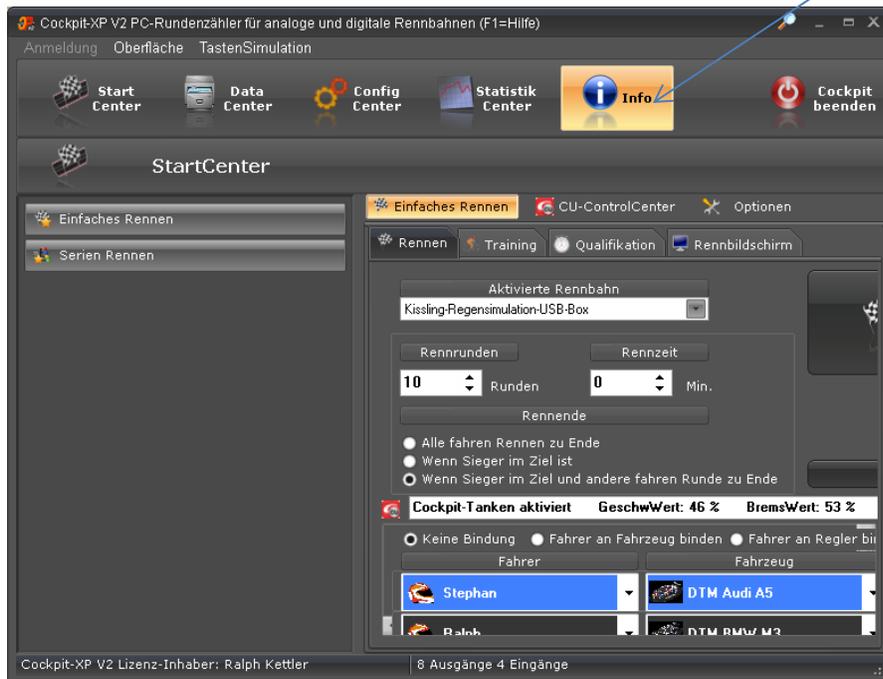
Vorraussetzungen für den Workshop

Bitte erst von meiner Website den Zip-Ordner **MeinErstesAddon.zip** runterladen und in das Anwender-Verzeichnis von Cockpit-XP kopieren und entpacken.

Wir brauchen diesen Ordner, da ich hier erkläre wie es grundsätzlich funktioniert...;-)....

Wo genau muss ich den Ordner hinkopieren?

Das richtige Verzeichnis bekommt man, wenn man hier auf Info klickt:



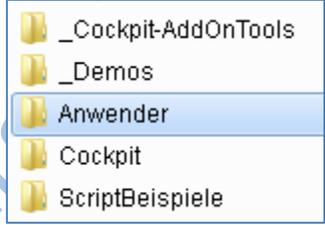
Dann öffnet sich dieses Fenster hier:



Hier steht dann das Zielverzeichnis

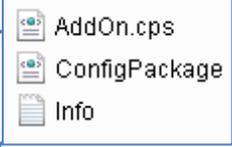
In diesem Ordner findet man dann den Ordner **AddOn** mit dem Unterordner **Anwender**.

	00_News	30.10.2014 21:09	Dateiordner
	01_Dokus	04.11.2014 18:41	Dateiordner
	AddOn	03.08.2014 16:56	Dateiordner
	Bilder	03.08.2014 16:56	Dateiordner
	Database	03.08.2014 16:56	Dateiordner
	Druckvorlagen	03.08.2014 16:56	Dateiordner
	Flaggen	03.08.2014 16:56	Dateiordner
	Languages	26.02.2013 22:39	Dateiordner
	Neue Version	25.10.2013 20:49	Dateiordner
	Online	03.08.2014 16:56	Dateiordner
	RennBildschirm	03.08.2014 16:56	Dateiordner
	Skins	03.08.2014 16:56	Dateiordner
	Sounds	03.08.2014 16:56	Dateiordner
	Sprachausgabe	07.02.2013 19:25	Dateiordner
	Treiber	30.10.2014 21:02	Dateiordner
	ZIP	14.03.2013 21:17	Dateiordner
	Cockpit-Licenc	15.02.2013 14:59	XML-Dokument
	FAQ	10.01.2013 15:41	HTML-Dokument
	License	10.01.2013 15:40	Textdokument
	LogAddOn	23.12.2013 20:31	Textdokument
	LogSystem	23.12.2013 20:35	Textdokument
	Manual	10.01.2013 15:41	HTML-Dokument
	PDOXUSRS.NET	04.11.2014 18:36	NET-Datei
	Version	11.01.2013 00:40	Textdokument



Hier dann den kompletten entpackten Ordner **MeinErstesAddon** reinkopieren.

	_LeeresPackage
	Archiv
	MeinErstesAddon
	RestrundenV2
	RK-DTM-6S-V10
	RK-Formel-1
	RK-RegenSimulationV3
	RK-Speedlimit-PSAU-E

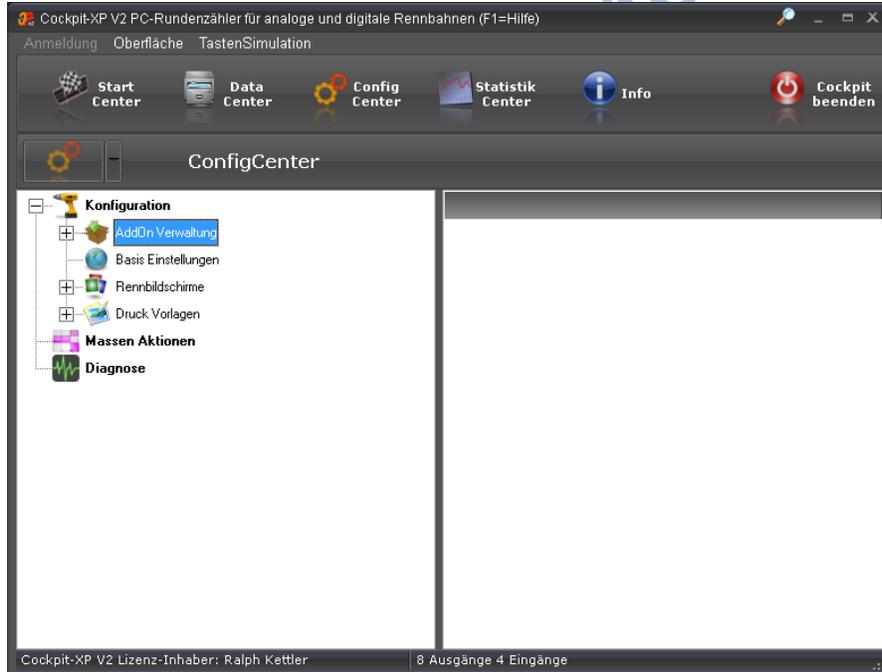


Ordner öffnen. Er sollte dann wie hier aussehen
Ist dies erledigt, dann können wir endlich loslegen.....

Als nächstes klicken wir auf **ConfigCenter**

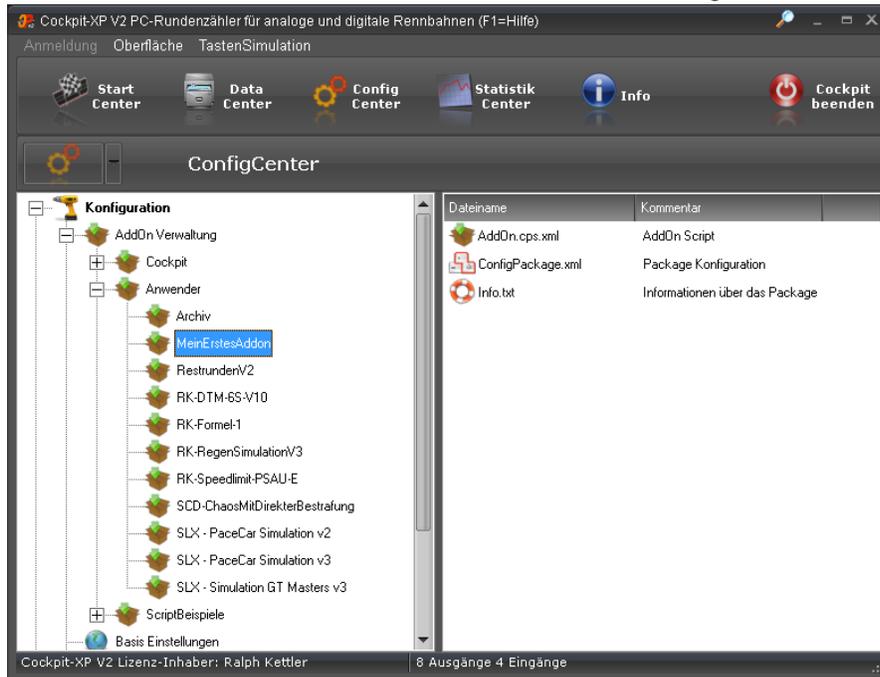


dann auf **AddOn Verwaltung** klicken



Hier klicken wir dann auf **Anwender**.

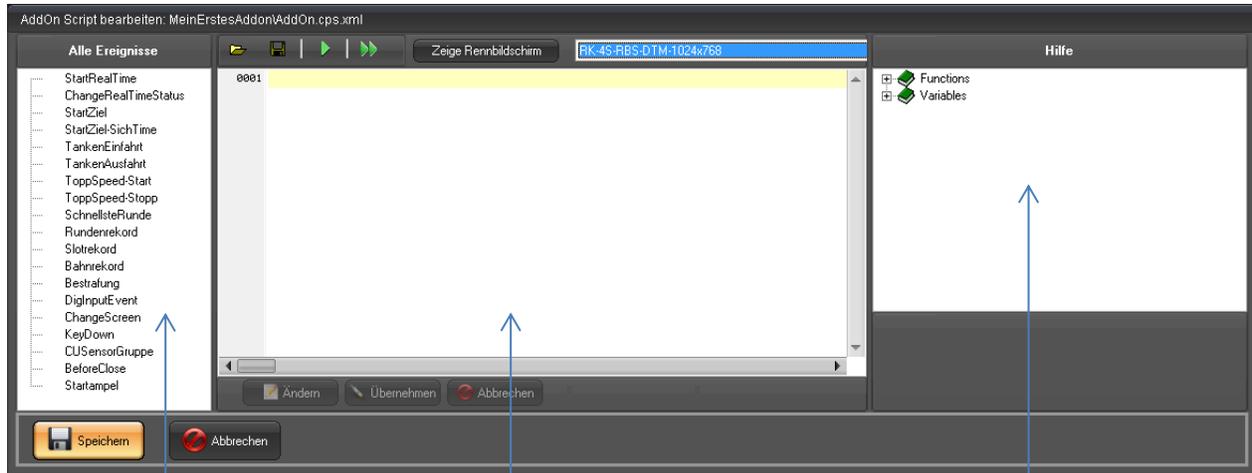
Hier sollte dann der Ordner **MeinErstesAddon** wie folgt zu sehen sein



Hier sieht man jetzt 3 Dateien:

- AddOn.cps.xml das ist das eigentliche „Programm“
- ConfigPackage.xml hier werden Variablen für die Konfig-Einstellungen erstellt, welche man dann später alle von „ausen“ einstellen kann
- Info.txt zusätzliche Informationen in Textform zum AddOn

Klickt man jetzt auf **AddOn.cps.xml**, dann öffnet sich die Programmierumgebung wie folgt:



<p>In der Spalte ganz links sehen wir verschiedene Begriffe. Diese Begriffe, nennen wir sie mal Events (Ereignisse) können einzeln angeklickt werden.</p>	<p>In der mittleren Spalte sehen wir eine 0001, das ist das eigentliche Editier-Fenster für das Programmieren</p>	<p>In der rechten Spalte gibt es die Begriffe Functions und Variables. Klickt man hier drauf, so öffnet sich eine weitere Auswahl an möglichen Funktionen und Variablen der „Programmiersprache“</p>
--	---	--

Was sind Ereignisse?

Für jedes dieser Ereignisse kann jeweils ein „Programm“ hinterlegt werden. Möchte man z.B. bei einem neuen **Rundenrekord** etwas ausgeben, dann klickt man auf **Rundenrekord** und programmiert dann dort ein „Programm“

Erklärung der einzelnen Ereignisse

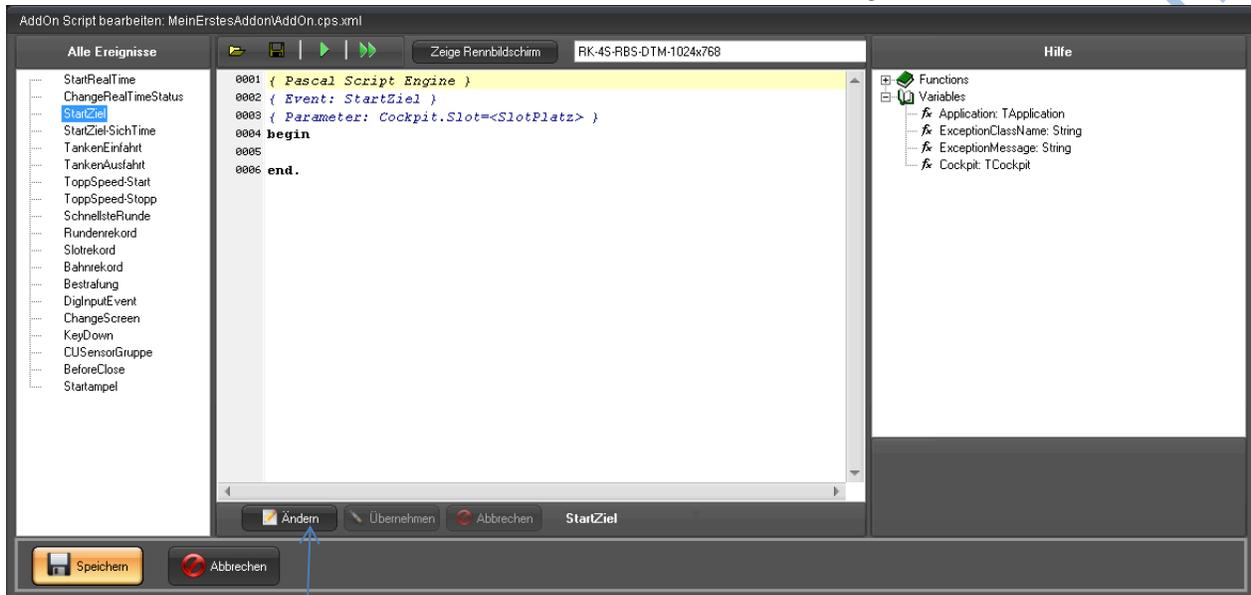
StartRealTime	
ChangeRealTimeStatus	
StartZiel	
StartZiel-SichTime	
TankenEinfahrt	
TankenAusfahrt	
ToppSpeedStart	
ToppSpeedStop	
SchnellsteRunde	
Rundenrekord	
Slotrekord	
Bahnrekord	
Bestrafung	
DigInputEvent	
ChangeScreen	
KeyDown	
CuSensorGruppe	
BeforeClose	
Startampel	

RK-Workshop-MeinErstesAddon

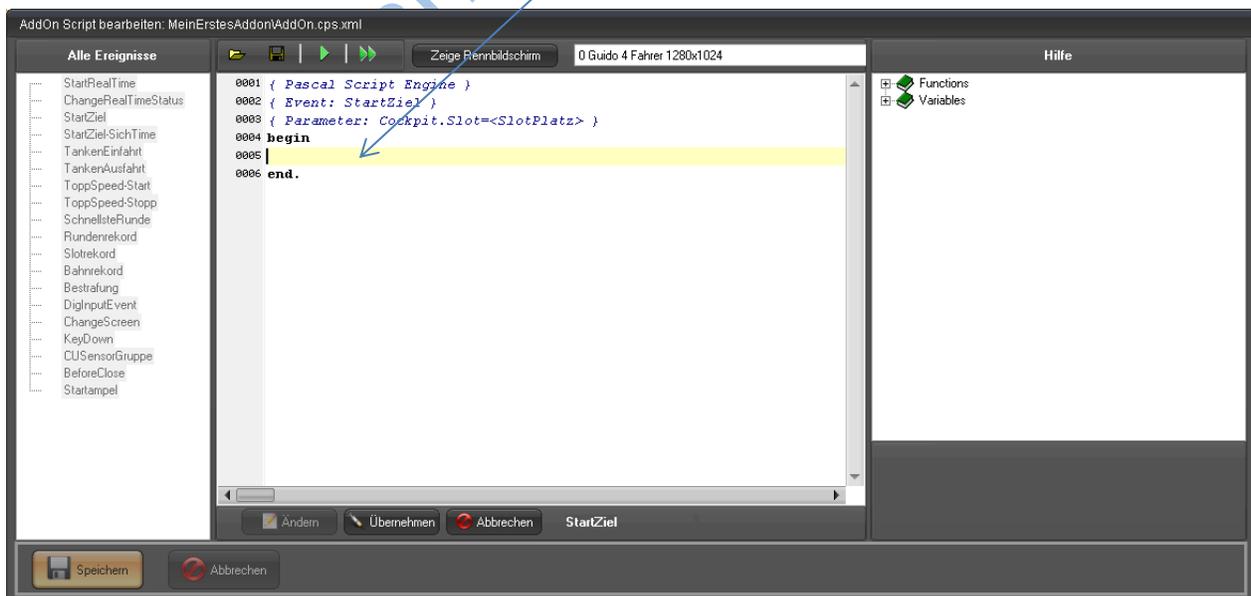
Was wollen wir programmieren?

Um mit der grundsätzlichen Funktion vertraut zu werden hab ich mir gedacht, wir programmieren mal einfach ein kleines AddOn, welches bei jeder Überfahrt Start/Ziel ansagt, welcher Fahrer gerade durchgefahren ist.

Dafür klicken wir mal auf **StartZiel**, dann öffnet sich in der Mitte folgendes Fenster:



Durch klicken auf **Ändern** können wir zwischen **begin** und **end** dann anfangen zu editieren. Der Cursor blinkt dann und die Zeile wird leicht gelb



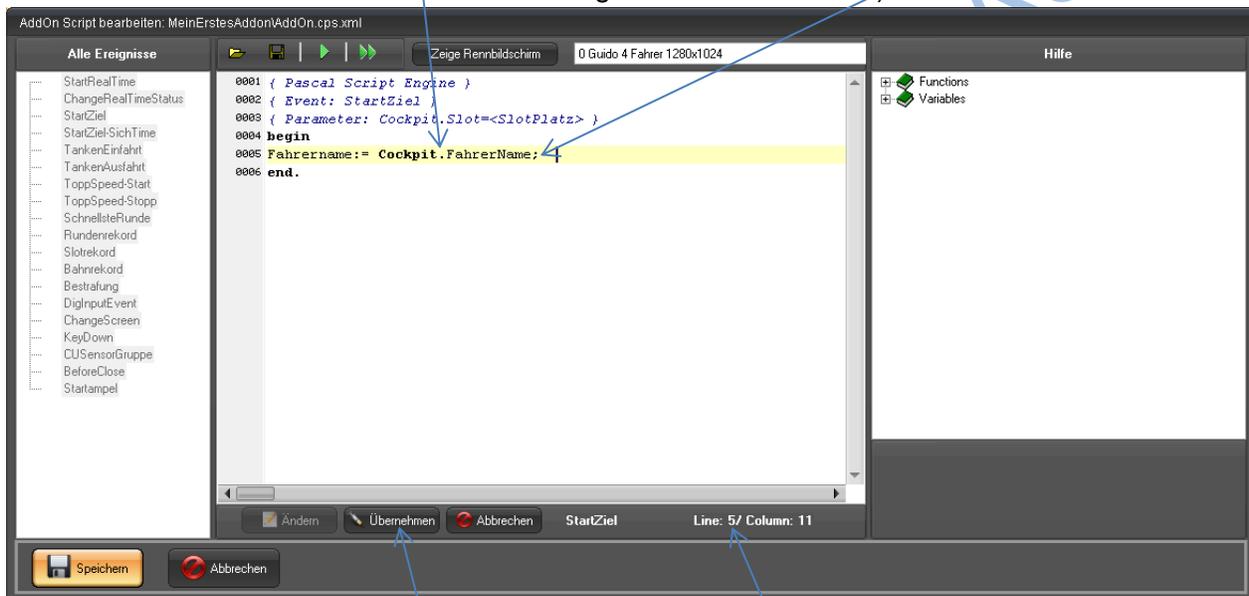
Wie bereits erwähnt arbeitet CXP das **StartZiel**-Ereignis ab, sobald es „angestossen“ wird. Woher soll aber die Software jetzt wissen, welcher Fahrer gerade über **StartZiel** gefahren ist? Um diese Info zu erhalten gibt es den -nennen wir es mal *Container*- Cockpit.Slot. Cockpit.Slot „enthält“ jetzt die Info, welche wir brauchen und zwar Cockpit.SlotID. Cockpit.SlotID ist der Verursacher, welche gerade über Start-Ziel gefahren ist.

So, jetzt wird es Ernst:

Wir schreiben jetzt den Namen des Fahrers in die Variable Fahrername.

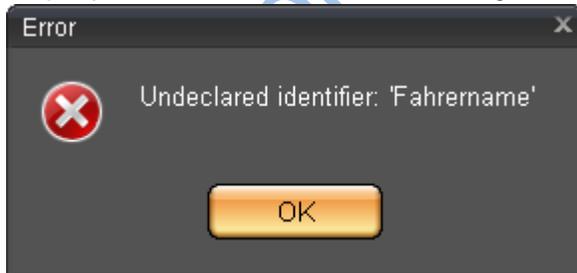
Dafür gegeben wir folgenden Programmcode ein:

Fahrername:=Cockpit.FahrerName und beenden den Eintrag mit einem Semikolon „;“ (braucht der Editor um sauber zu erkennen wo die Programmierzeile aufhört)



Jetzt klicken wir mal auf **Übernehmen**

Oups, jetzt kommt eine Fehlermeldung?!?!?



... P.S. hier wird zusätzlich angezeigt, in welcher Programmierzeile der Fehler gefunden wurde

Was ist passiert?

Der Editor checkt bei Übernahme auf Plausibilität wie z.B. ob alle Variablen sauber deklariert sind. In unserem Fall ist es nicht so. Wir müssen jetzt die Variable „**Fahrername**“ noch sauber deklarieren.

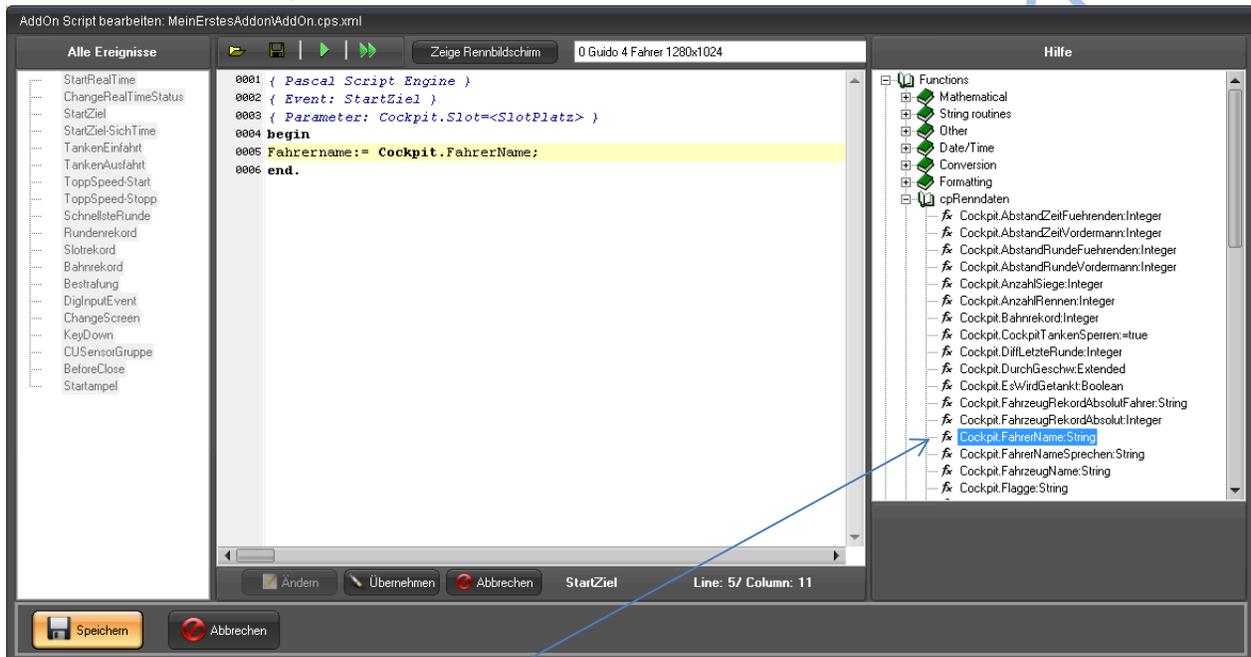
Variablendeklarationen

Grundsätzlich werden Variablen nach Programmzeile 0003 und vor der Programmzeile **begin** wie folgt deklariert:

Da es sich beim Fahrername um einen String handelt, muss man die Variable dementsprechend deklarieren.

Woher weiß ich den, um was für eine Art Variable es sich handelt?

Dafür klicken wir mal ganz rechts auf **Functions** und dann auf **cpRenndaten**



Dann hier nach **Cockpit.FahrerName** suchen und dort steht nach dem Doppelpunkt „:“ um welche Art von Variable es sich handelt, in unserem Fall also um einen String.....

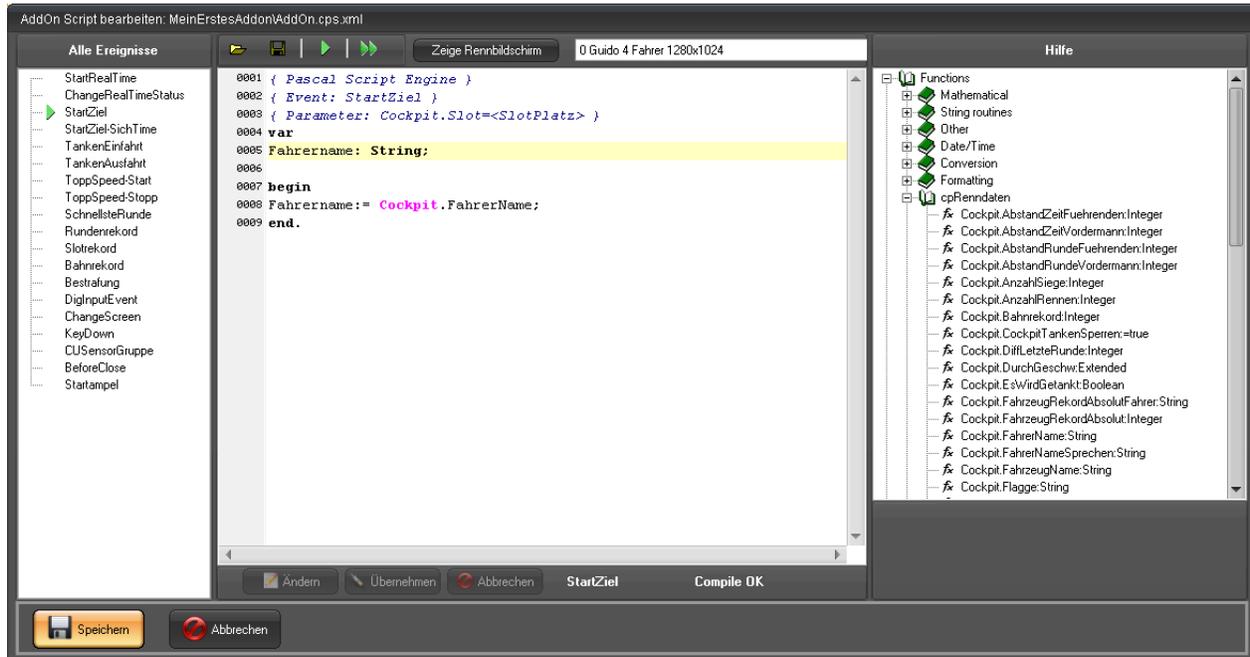
→ also, jetzt deklarieren wir diesen Variablennamen als String wie folgt:Angabe Namen, dann ein Doppelpunkt „:“ und die Art der Variablen, abgeschlossen mit einem Semikolon

```

var
  Fahrername: String;

```

Die unbedingt vor **begin** rein-editieren !!



Drückt man jetzt auf Übernehmen, kommt keine Fehlermeldung mehr.

Was haben wir bis jetzt gelernt?

- Wir können den AddOn-Editor aufrufen
- Wir wissen jetzt warum und wie man Variablen deklariert
- Wir wissen jetzt, dass wir jede Programmierzeile mit einem Semikolon abschließen müssen

Weiter geht's.

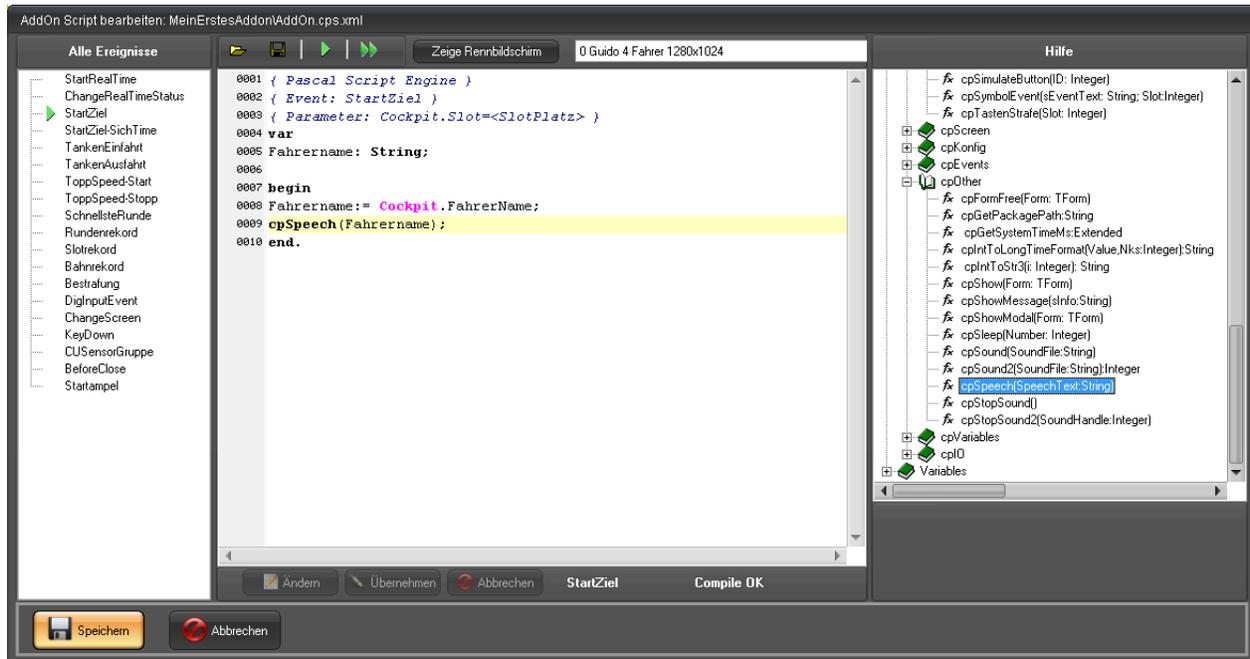
Bisher haben wir ja nur den richtigen Namen in die Variable geschrieben, wir wollen ihn aber ja hören bei Überfahrt Start/Ziel. Dafür gibt es den Befehl **cpSpeech(SpeechText:String)**

Kurze Einleitung in die Befehls-Struktur

Vorne steht der „Befehl“, gefolgt von einer Angabe in Klammer (). Innerhalb der Klammer wird noch angegeben, welche Variablen-Art benötigt wird. In unserem Fall brauchen wir einen String. Dieser String muss innerhalb der zwei Klammern stehen.

Somit brauchen wir noch für die Ausgabe des Fahrername folgende Befehlszeile:

`cpSpeech(Fahrername);`



So sollte es dann aussehen und schon sind wir (fast) fertig.....

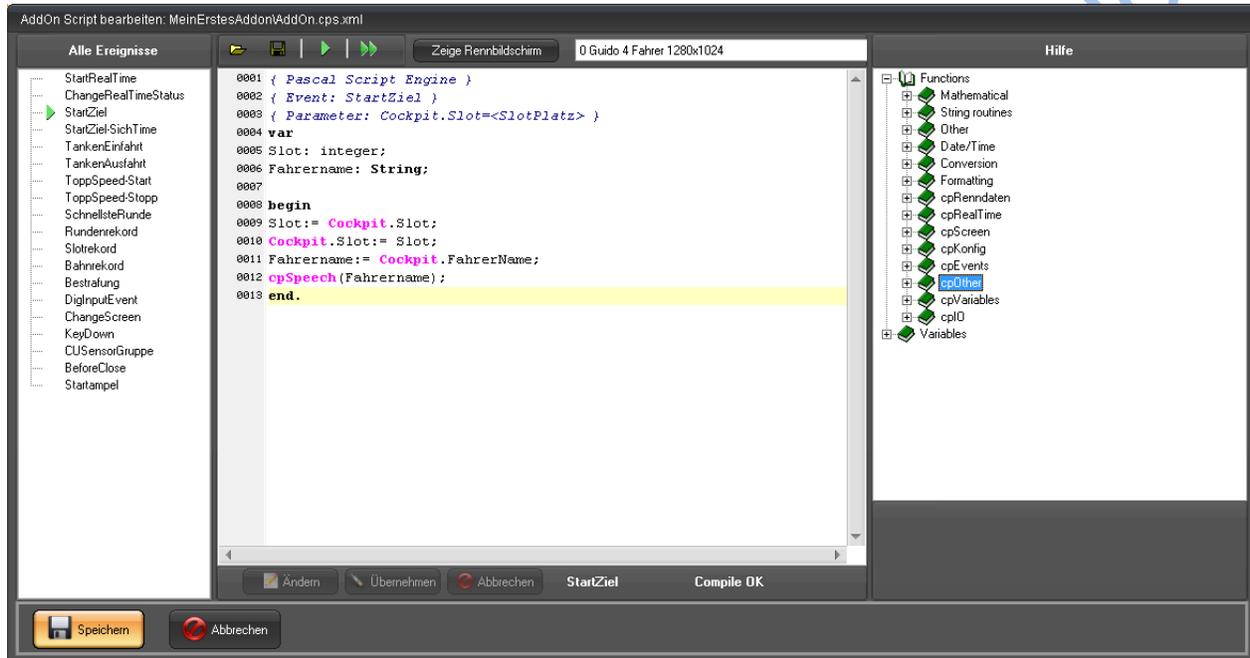
Es gibt eine Eigenart bei Cockpit XP, welche mit folgenden zwei Programmzeilen behoben wird:

Slot:=Cockpit.Slot;

Cockpit.Slot:= Slot;

Nicht fragen was das soll....wird später noch ausführlich erklärt.....

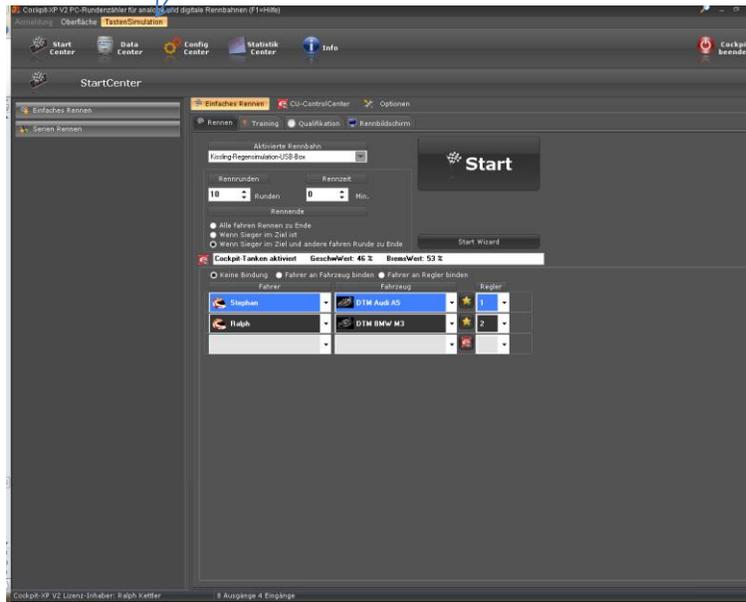
Also bitte einfach diese zwei Programmzeilen noch direkt nach **begin** einfügen, inkl. Der Variablendeklaration **Slot: integer;**



So, jetzt ist unser kleines Programm endlich fertig.....;-).....jetzt mal auf Speichern klicken....

Test von AddOns

AddOns kann man natürlich direkt an der Bahn testen. Es gibt aber auch eine Möglichkeit über die **Tastenfunktion** ein AddOn am Rechner (ohne Rennbahn) anzutesten.



Hier mal draufklicken, dann kann man die Tastensimulation aktivieren (dabei wird dann vor aktivieren ein Häkchen gesetzt)

Jetzt noch geschwind das AddOn auch laden, einen RBS auswählen auf Start drücken und simulieren.

Dafür gibt es dann zusätzlich zum RBS noch folgendes neues Fenster:

	<p>Hier sieht man die USBBox, wenn in der Bahn konfiguriert und die CU</p>
	<p>Diese Felder hier sind die Eingänge bzw. die Ausgänge der USBBox. Nennen wir diese Feld ab sofort USBBox-Feld</p>
	<p>Diese Felder hier sind für die CU: E1 = Überfahrt Start/Ziel E6 = TankenEinfahrt E7 = TankenAusfahrt</p>
	<p>Hier wählt man aus, mit welcher ID man simulieren möchte. In diesem Fall ist es die ID 2</p>

Wie simuliert man jetzt?

- erst einmal muss ein Training/Quali oder Rennen gestartet werden
- dann wählt man die entsprechende ID aus im ID-Feld
- will man eine Start/Ziel-Durchfahrt simulieren so klickt man auf E1 im CU-Feld
- will man TankenEinfahrt simulieren, dann klickt man auf E6 im CU-Feld
- will man TankenAusfahrt simulieren, dann klickt man auf E7 im CU-Feld

Mit unserem kleinen Addon müsste jetzt bei jedem Klick auf E1 im CU-Feld der Name des Fahrers angesagt werden.

RK-Workshop-MeinErstesAddOn

Impressum

Sollte jemand zufällig Gefallen gefunden haben an meinem Workshop so darf er ruhig sein Wohlgefallen in Form einer kleinen freiwillige Zuwendungen für meinen extremen Bier- und Kaffeeverbrauchs während der Erstellung dieser Doku zukommen lassen. Hier eine von vielen Möglichkeiten:

http://www.amazon.de/gp/registry/wishlist/3B76OAKWIN7X9/ref=cm_sw_em_r_wsg_bYs1sb091G4XC_wp

RK-Workshop-MeinErstesAddon